3D transformations and hierarchical modeling

From 2D to 3D

- Much +/- the same:
  - Translation, scaling
  - Homogeneous vectors: one extra coordinate
  - Matrices: 4x4
- Rotation more complex

Transformation in 3D

- Transformation Matrix

$$\begin{bmatrix} A & D & G & J \\ B & E & H & K \\ C & F & I & L \\ 0 & 0 & 0 & S \end{bmatrix} \rightarrow \left[\begin{array}{c|c} 3\times 3 & 3\times 1 \\ \hline 1\times 3 & 1\times 1 \end{array}\right]$$

3×3 : Scaling, Reflection, Shearing, Rotation

3×1 : Translation

1×1 : Uniform global Scaling

1×3 : Homogeneous representation

3D Translation 1

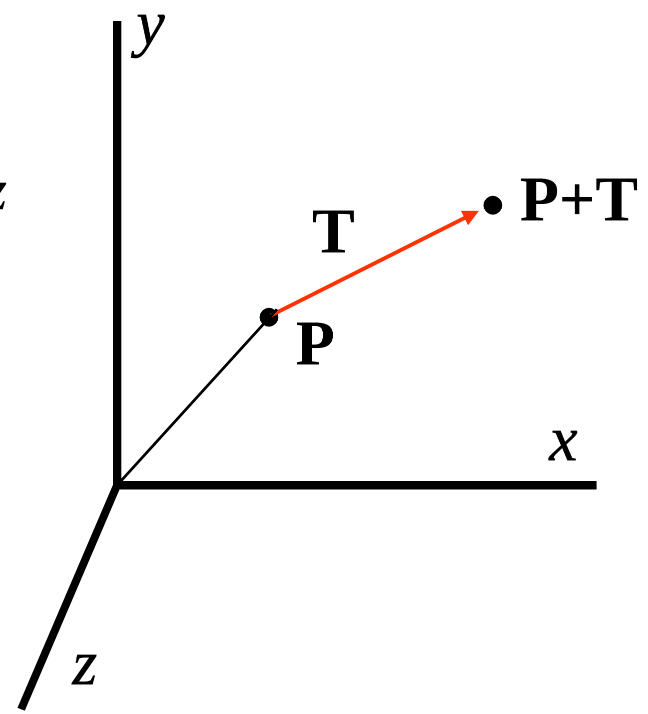
Translate over vector (t_x, t_y, t_z):

$$x' = x + t_x,\quad y' = y + t_y\,,\quad z' = z + t_z$$

or

$$\mathbf{P}' = \mathbf{P} + \mathbf{T},\ \text{met}$$

$$\mathbf{P}' = \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix}, \mathbf{P} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \text{en } \mathbf{T} = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}$$

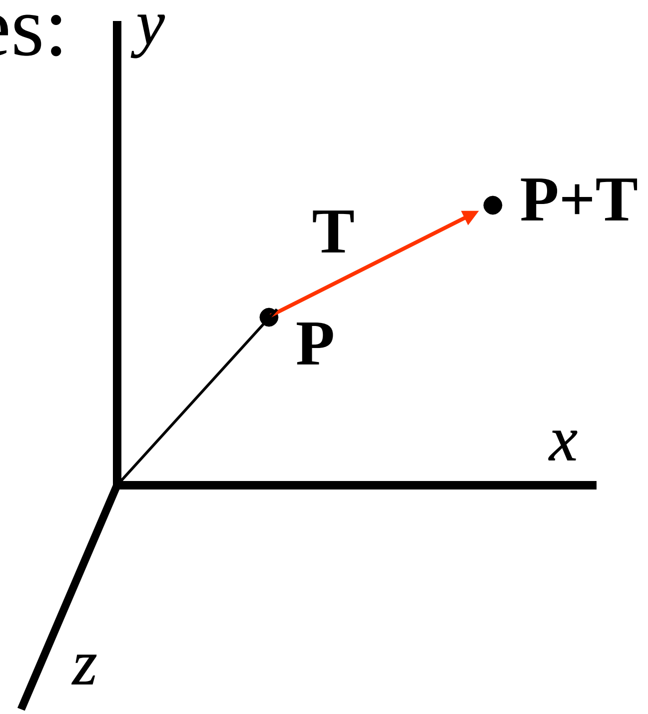


3D Translation 2

In 4D homogeneous coordinates:

$$\mathbf{P}' = \mathbf{MP}, \text{ or}$$

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$



3D Rotation 1

$$\text{Rotate over angle } \alpha \text{ around } z\text{-as:}$$

$$x' = x\cos\alpha - y\sin\alpha$$

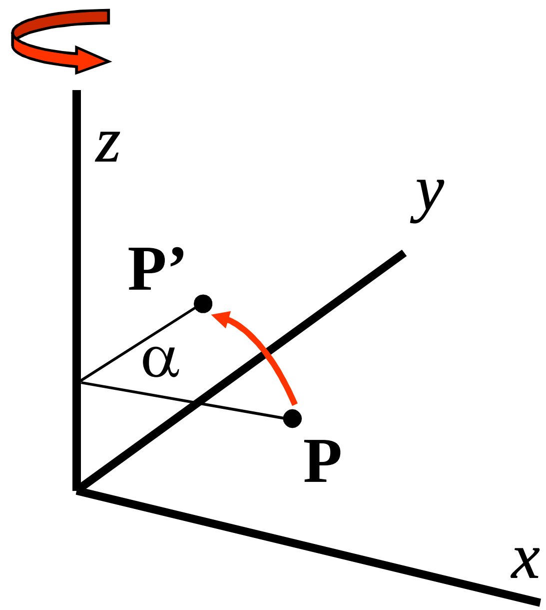
$$y' = x\sin\alpha + y\cos\alpha$$

$$z' = z$$

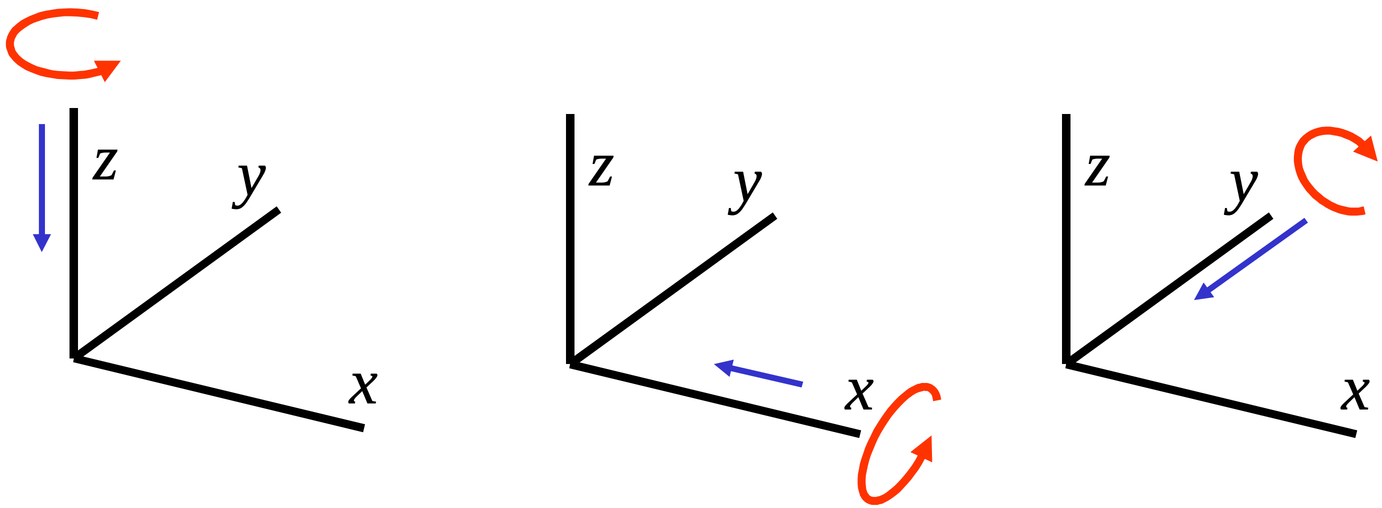
Or

$$\mathbf{P}' = \mathbf{R}_z(\alpha)\mathbf{P}, \ \text{with}$$

$$\mathbf{R}_z(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



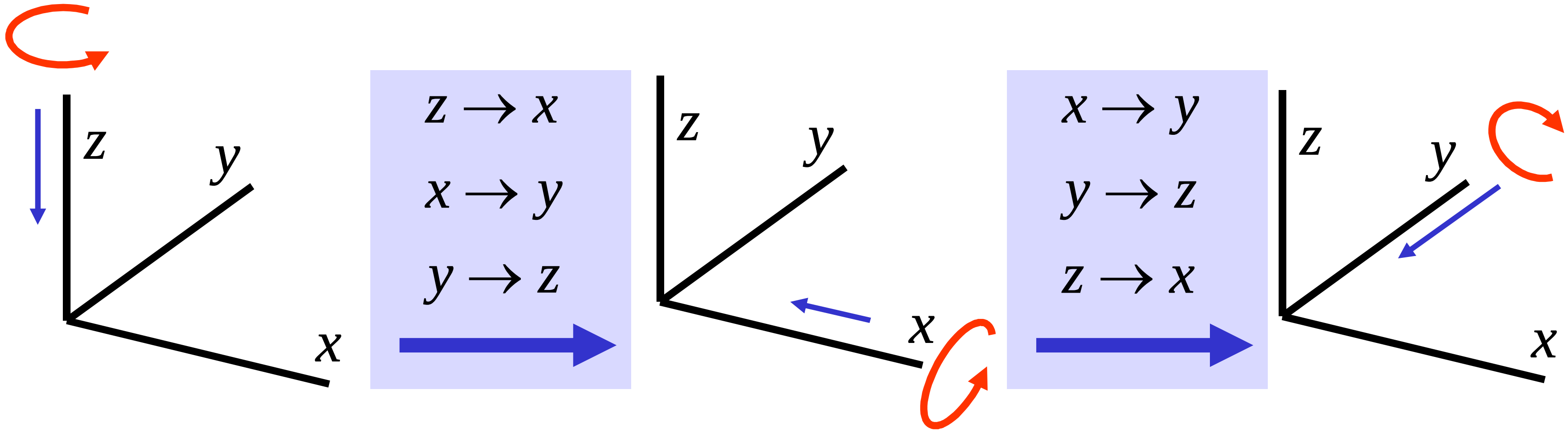
3D Rotation 2



Rotation around axis:

- Counterclockwise, viewed from rotation axis

3D Rotation 3



Rotation around axes:

Cyclic permutation coordinate axes

$x \to y \to z \to x$

3D Rotatie 4

Rotate over angle α around z-as:

$x' = x\cos\alpha - y\sin\alpha$

$y' = x\sin\alpha + y\cos\alpha$

$z' = z$

Or

$\mathbf{P}' = \mathbf{R}_z(\alpha)\mathbf{P}$, with

$$\mathbf{R}_z(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$z \rightarrow x$

$x \rightarrow y$

$y \rightarrow z$

Rotate over angle α around x-as:

$y' = y\cos\alpha - z\sin\alpha$

$z' = y\sin\alpha + z\cos\alpha$

$x' = x$

Or

$\mathbf{P}' = \mathbf{R}_x(\alpha)\mathbf{P}$, with

$$\mathbf{R}_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

3D Rotation around arbitrary axis 1

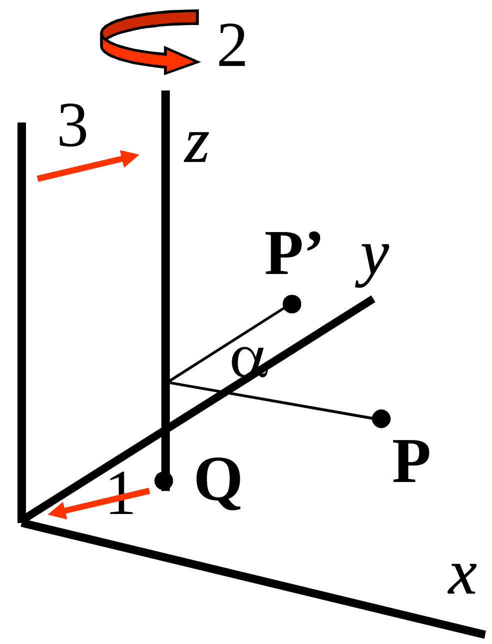
Rotation around axis, parallel to coordinate axis, through point **Q**.

For example. the z - as. Similar as 2D rotation :

1. Translate over **-Q**;
2. Rotate around z - axis;
3. Translate back over **Q**.

Or :

$$\mathbf{P}' = \mathbf{T}(\mathbf{Q})\mathbf{R}_{\mathbf{z}}(\alpha)\mathbf{T}(-\mathbf{Q})\mathbf{P}$$

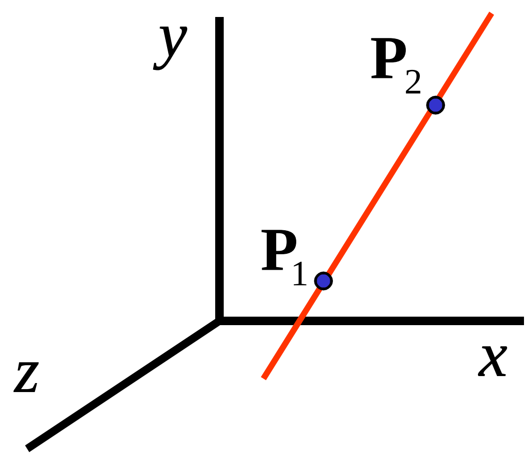


3D Rotation around arbitrary axis 2

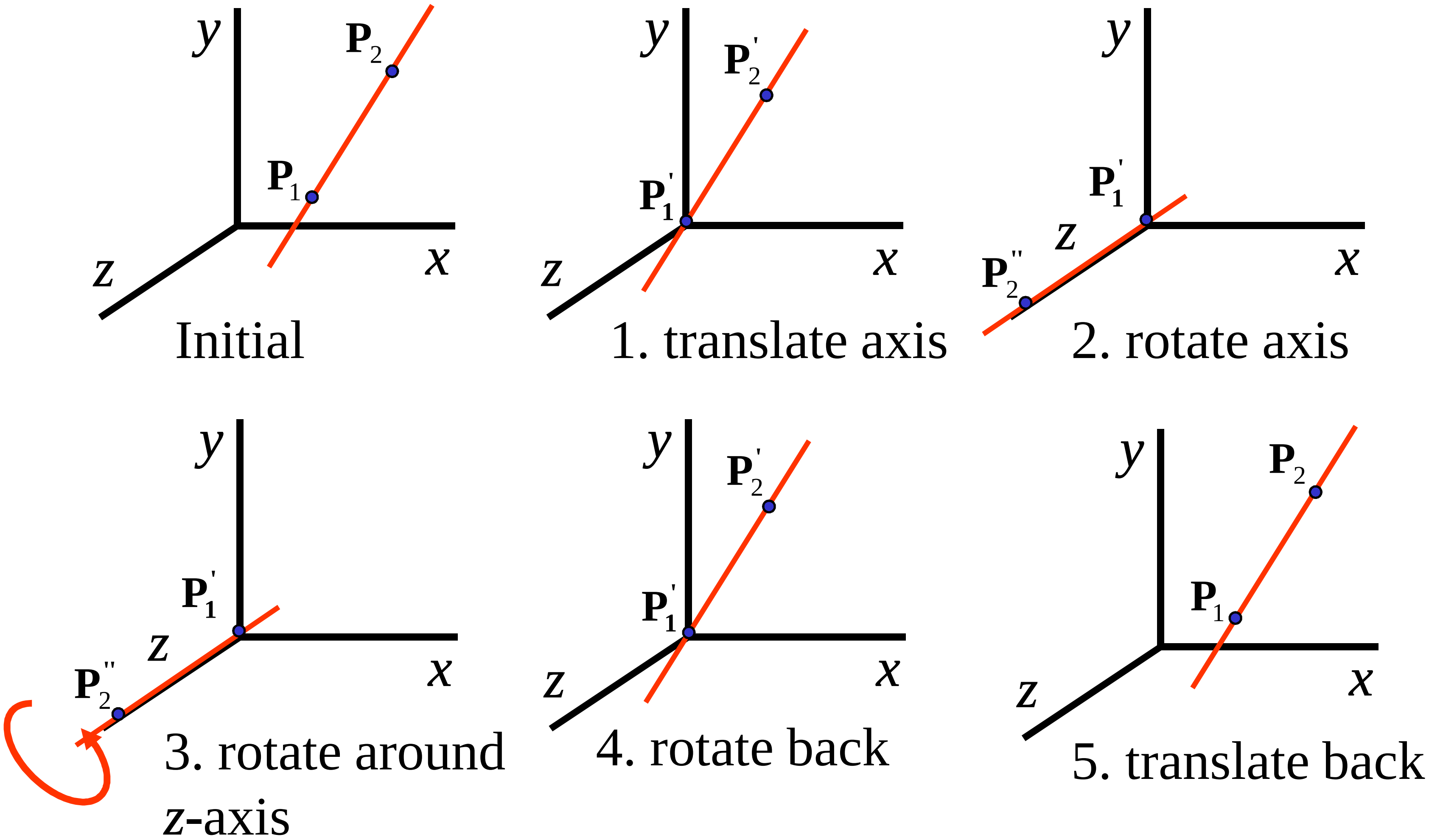
Rotation around axis through two points $\mathbf{P}_1$ and $\mathbf{P}_1$.

More complex:

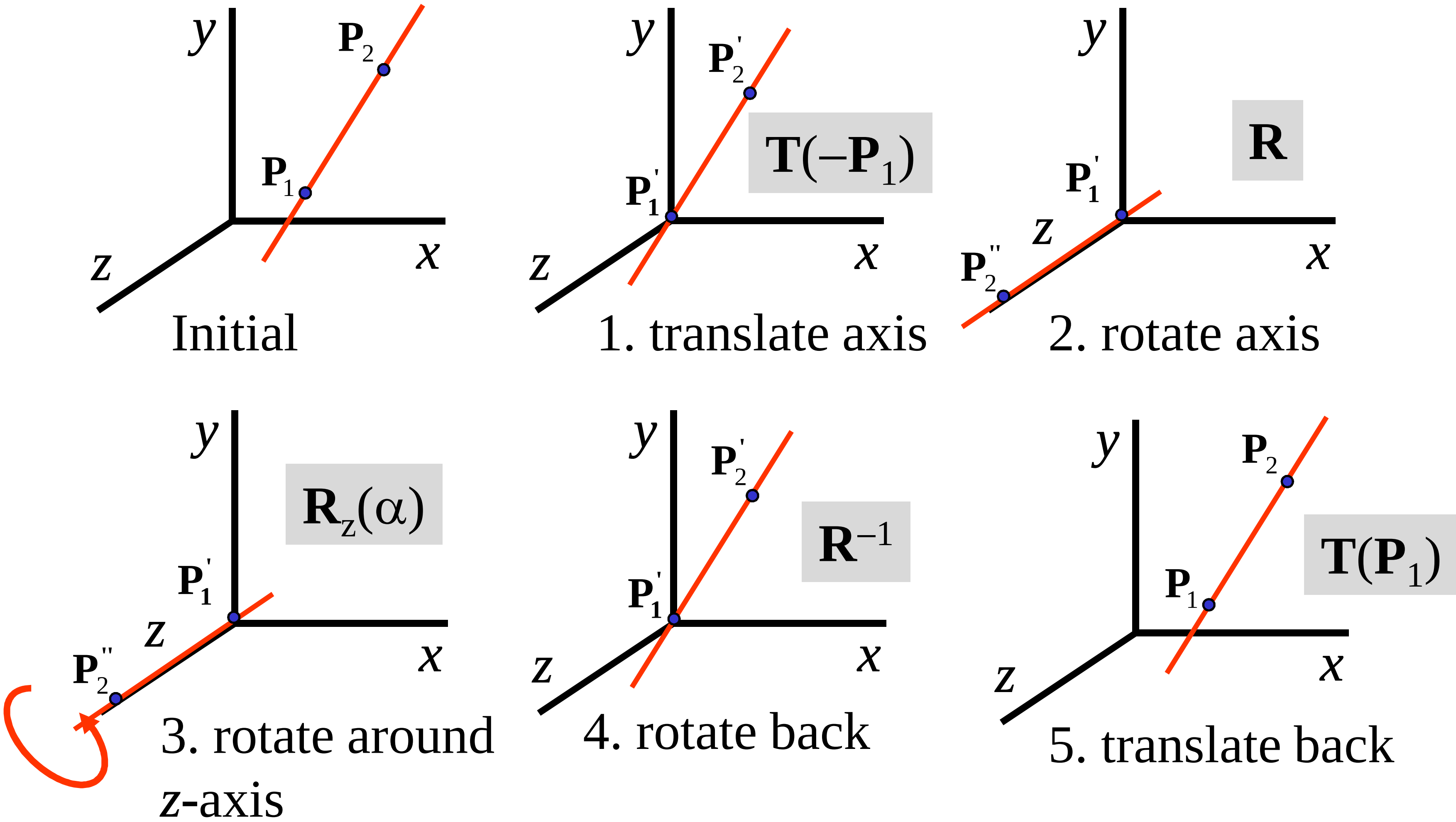
1. Translate such that axis goes through origin;
2. Rotate…
3. Translate back again.



3D Rotation around arbitrary axis 3

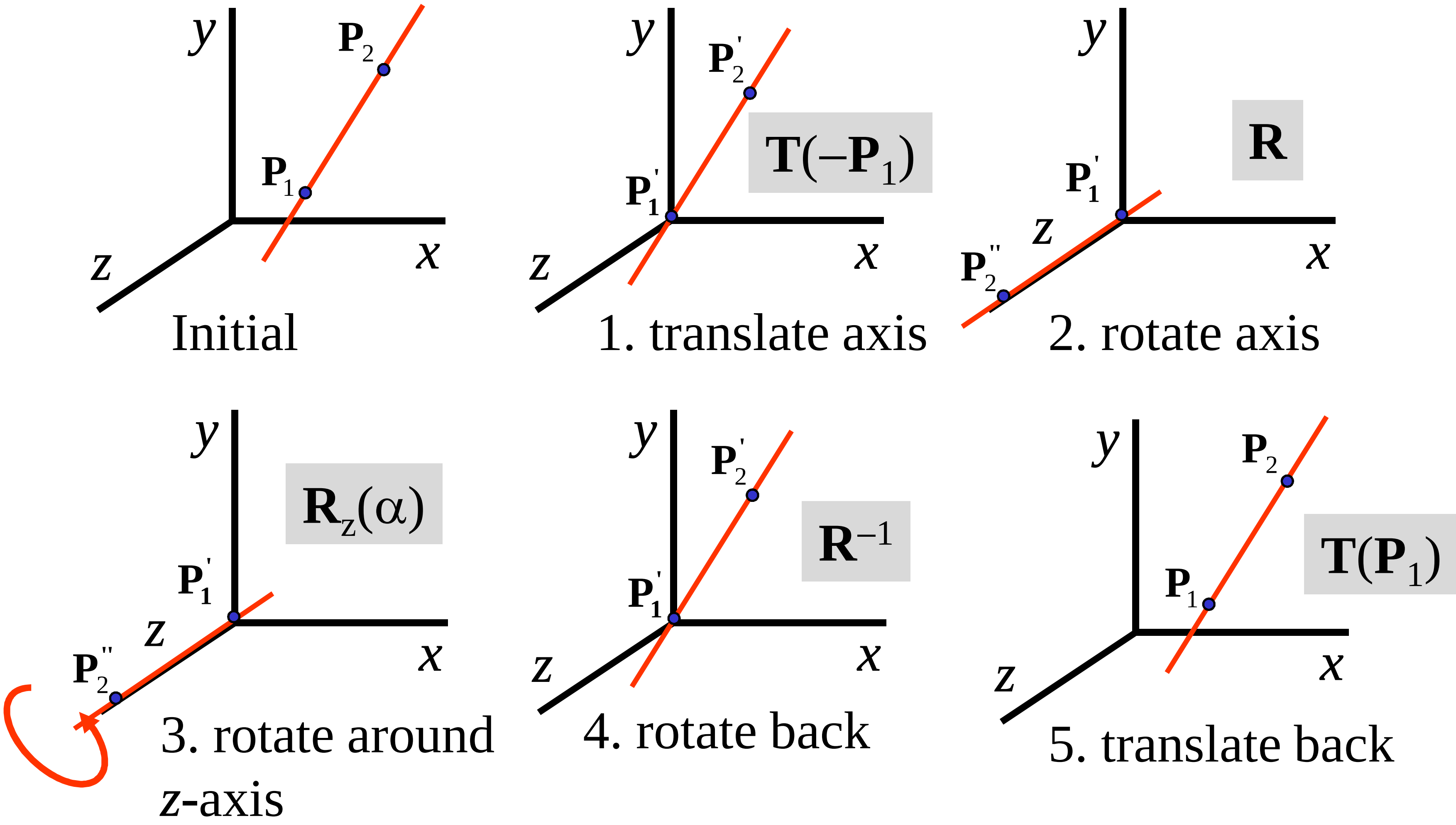


3D Rotation around arbitrary axis 3



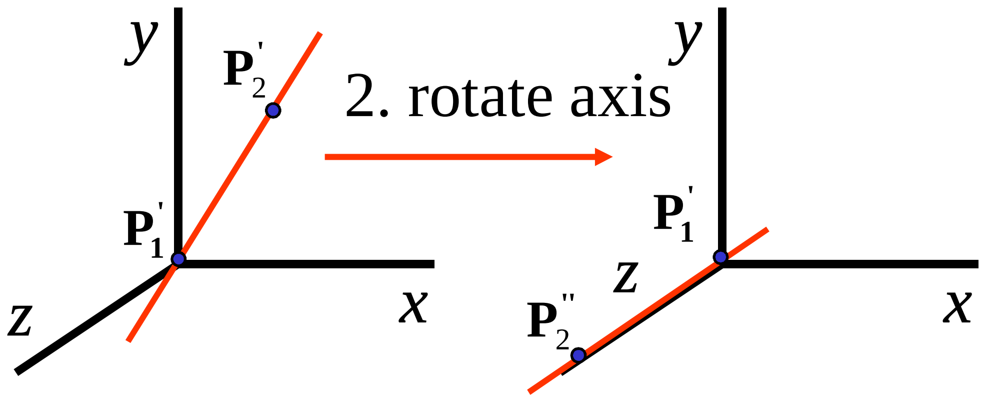
3D Rotation around arbitrary axis 3

$$\mathbf{M} = \mathbf{T}(\mathbf{P}_1)\ \mathbf{R}^{-1}\mathbf{R}_z(\alpha)\ \mathbf{R}\mathbf{T}(-\mathbf{P}_1)$$



3D Rotation around arbitrary axis 4

Difficult step: **R**



Find rotation such that rotation axis aligns with *z*-axis.

Two options:

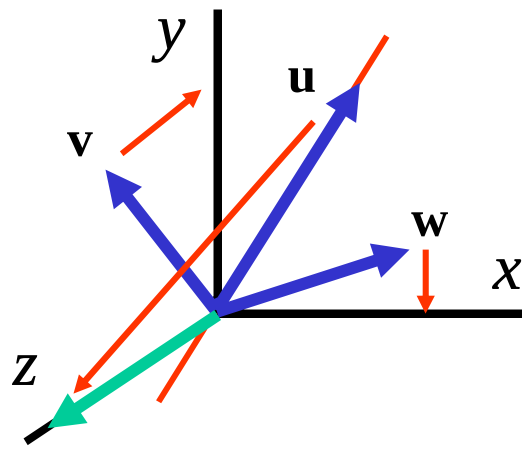
1. Step by step by step
2. Direct derivation of matrix

3D Rotation around arbitrary axis 5

1. Construct orthonormal axis frame **u**, **v**, **w**;

2. Invent rotation matrix **R,** such that:

u is mapped to *z*-axis;
v is mapped to *y*-axis;
w is mapped to *x*-axis.



3D Rotation around arbitrary axis 6

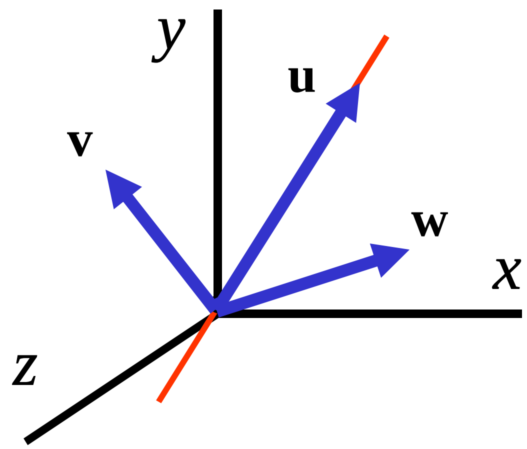
Construct orthonormal axis frame **u**, **v**, **w**:

$\mathbf{u} = (\mathbf{P}_2 - \mathbf{P}_1) \ / \ |\mathbf{P}_2 - \mathbf{P}_1|$

$\mathbf{v} = \mathbf{u} \times (1,0,0) \ / \ | \ \mathbf{u} \times (1,0,0) \ |$

$\mathbf{w} = \mathbf{v} \times \mathbf{u}$

(If **u** = (a, 0, 0), then use (0, 1, 0))



This frame is orthonormal:

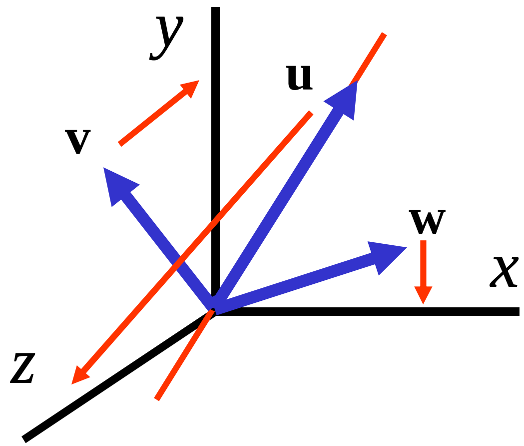
Unit length axes: **u.u** = **v.v** = **w.w** = 1

All axes perpendicular: **u.v** = **v.w** = **w.u** = 0

3D Rotation around arbitrary axis 7

Generic rotation matrix :

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



Fill in **R** such that

$$\boxed{\mathbf{R}\begin{pmatrix} u_x \\ u_y \\ u_z \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}}, \quad \boxed{\mathbf{R}\begin{pmatrix} v_x \\ v_y \\ v_z \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}}, \quad \text{and} \quad \boxed{\mathbf{R}\begin{pmatrix} w_x \\ w_y \\ w_z \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}}$$

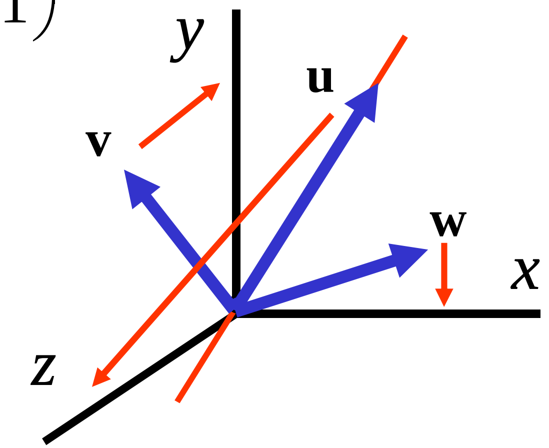
3D Rotation around arbitrary axis 8

Fill **R**, such that

$$\mathbf{R}\begin{pmatrix} u_x \\ u_y \\ u_z \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \quad \mathbf{R}\begin{pmatrix} v_x \\ v_y \\ v_z \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}, \quad \text{en ind } \mathbf{R}\begin{pmatrix} w_x \\ w_y \\ w_z \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Solution :

$$\mathbf{R} = \begin{pmatrix} \boxed{w_x \quad w_y \quad w_z} & 0 \\ \boxed{v_x \quad v_y \quad v_z} & 0 \\ \boxed{u_x \quad u_y \quad u_z} & 0 \\ 0 \quad 0 \quad 0 & 1 \end{pmatrix}$$

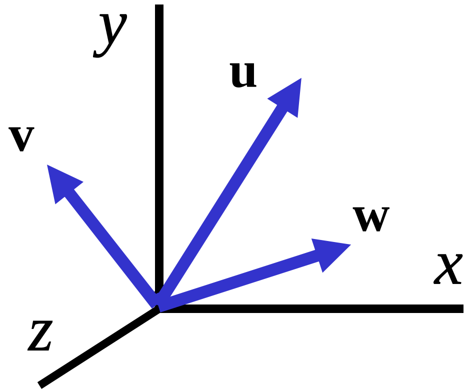


Done! But how to find $\mathbf{R}^{-1}$?

Inverse of rotation matrix 1

Each rotation matrix is an orthonormal matrix **M**:

$$\mathbf{M} = \begin{pmatrix}\mathbf{u} & \mathbf{v} & \mathbf{w}\end{pmatrix} = \begin{pmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{pmatrix}$$



The frame **u**, **v**, **w** is orthonormal:

Unit length axes: **u.u** = **v.v** = **w.w** = 1

All axes perpendicular: **u.v** = **v.w** = **w.u** = 0.

Requested: $\mathbf{M}^{-1}$ such that $\mathbf{M}^{-1}\mathbf{M} = \mathbf{I}$

Inverse of rotation matrix 2

Requested: $\mathbf{M}^{-1}$ such that $\mathbf{M}^{-1}\mathbf{M} = \mathbf{I}$

Solution:

$$\mathbf{M}^{-1} = \mathbf{M}^{T} = \begin{pmatrix} \mathbf{u}^{T} \\ \mathbf{v}^{T} \\ \mathbf{w}^{T} \end{pmatrix} = \begin{pmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ w_x & w_y & w_z \end{pmatrix}$$

In words:

The inverse of a rotation matrix is the transpose.

(For a rotation around the origin).

Inverse of rotation matrix 3

Requested: $\mathbf{M}^{-1}$ such that $\mathbf{M}^{-1}\mathbf{M} = \mathbf{I}$

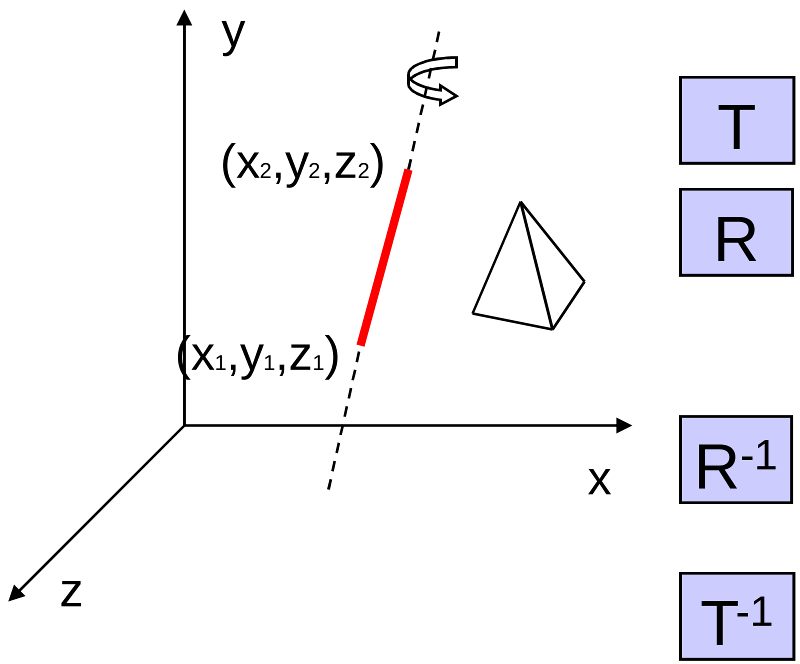
Solution: $\mathbf{M}^{-1} = \mathbf{M}^{T}$

Check:

$$\mathbf{M}^T\mathbf{M} = \begin{pmatrix} \mathbf{u}^T \\ \mathbf{v}^T \\ \mathbf{w}^T \end{pmatrix} \begin{pmatrix} \mathbf{u} & \mathbf{v} & \mathbf{w} \end{pmatrix} = \begin{pmatrix} \mathbf{u}\cdot\mathbf{u} & \mathbf{u}\cdot\mathbf{v} & \mathbf{u}\cdot\mathbf{w} \\ \mathbf{v}\cdot\mathbf{u} & \mathbf{v}\cdot\mathbf{v} & \mathbf{v}\cdot\mathbf{w} \\ \mathbf{w}\cdot\mathbf{u} & \mathbf{w}\cdot\mathbf{v} & \mathbf{w}\cdot\mathbf{w} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \mathbf{I}$$

General 3D Rotations

- Rotation about an Arbitrary Axis



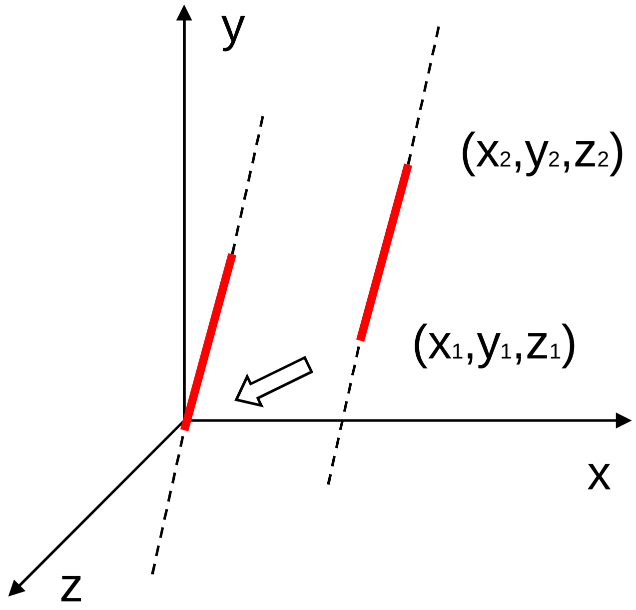
Basic Idea

1. **Translate (x1, y1, z1) to the origin**
2. **Rotate (x'2, y'2, z'2) on to the z axis**
3. **Rotate the object around the z-axis**
4. **Rotate the axis to the original orientation**
5. **Translate the rotation axis to the original position**

$$\mathbf{R}(\theta)=\mathbf{T}^{-1}\mathbf{R}_x^{-1}(\alpha)\mathbf{R}_y^{-1}(\beta)\mathbf{R}_z(\theta)\mathbf{R}_y(\beta)\mathbf{R}_x(\alpha)\mathbf{T}$$

General 3D Rotations

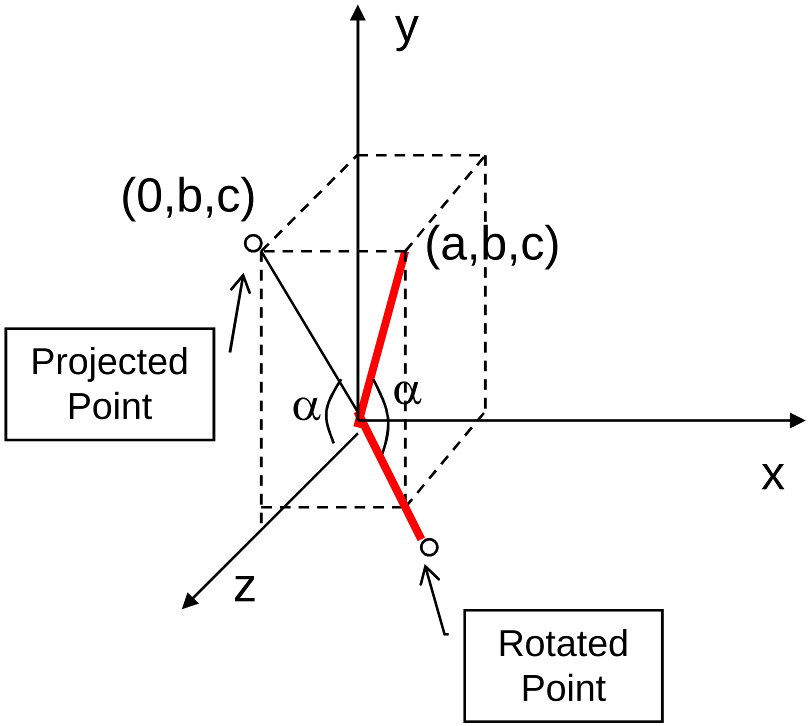
- Step 1. Translation



$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & -x_1 \\ 0 & 1 & 0 & -y_1 \\ 0 & 0 & 1 & -z_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

General 3D Rotations

- Step 2. Establish $[\,T_R\,]^{\alpha}_{x}$ x axis



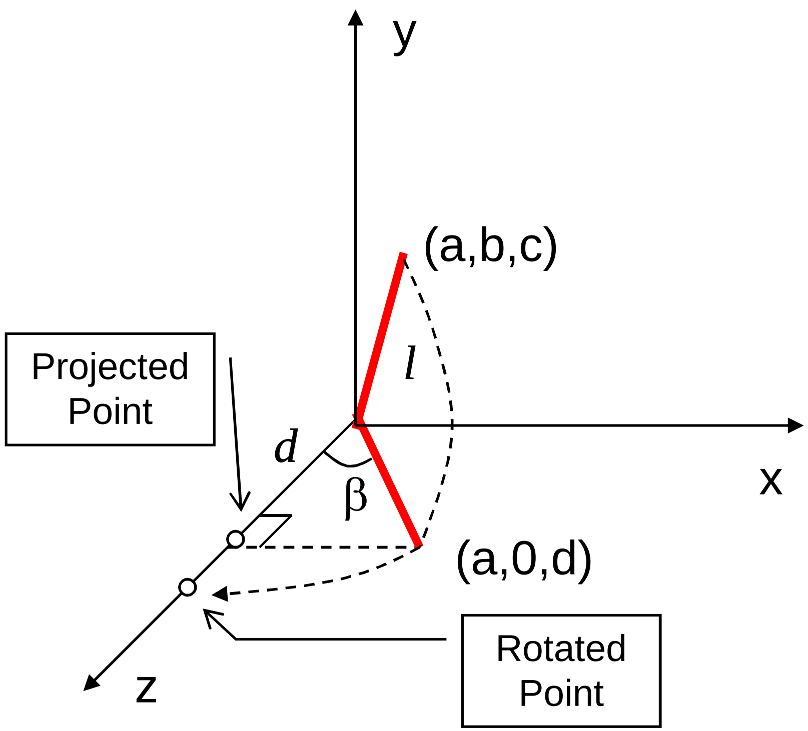
$$\sin\alpha = \frac{b}{\sqrt{b^2+c^2}} = \frac{b}{d}$$

$$\cos\alpha = \frac{c}{\sqrt{b^2+c^2}} = \frac{c}{d}$$

$$\mathbf{R}_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c/d & -b/d & 0 \\ 0 & b/d & c/d & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Arbitrary Axis Rotation

- Step 3. Rotate about *y* axis by ϕ



$$\sin\beta = \frac{a}{l}, \qquad \cos\beta = \frac{d}{l}$$

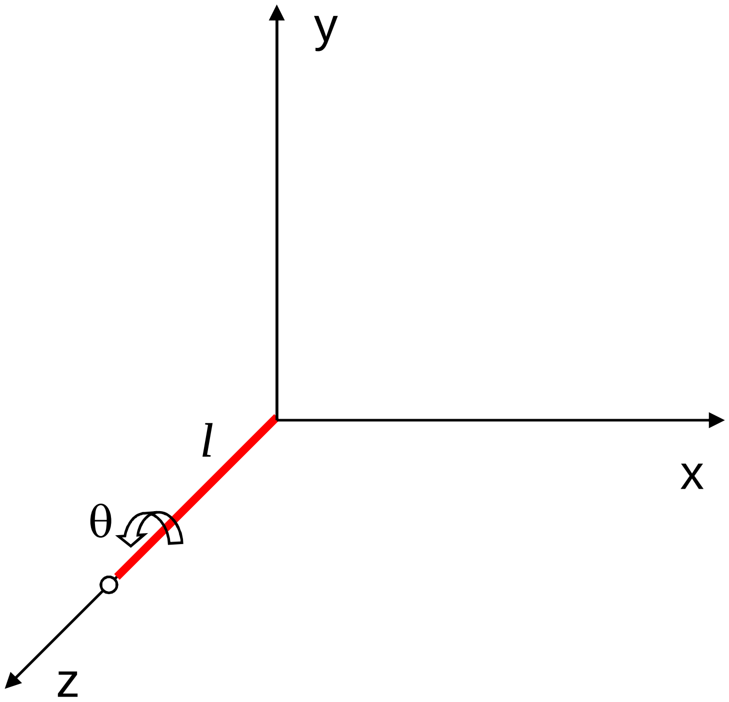
$$l^2 = a^2 + b^2 + c^2 = a^2 + d^2$$

$$d = \sqrt{b^2 + c^2}$$

$$\mathbf{R}_y(\beta) = \begin{bmatrix} \cos\beta & 0 & -\sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} d/l & 0 & -a/l & 0 \\ 0 & 1 & 0 & 0 \\ a/l & 0 & d/l & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Arbitrary Axis Rotation

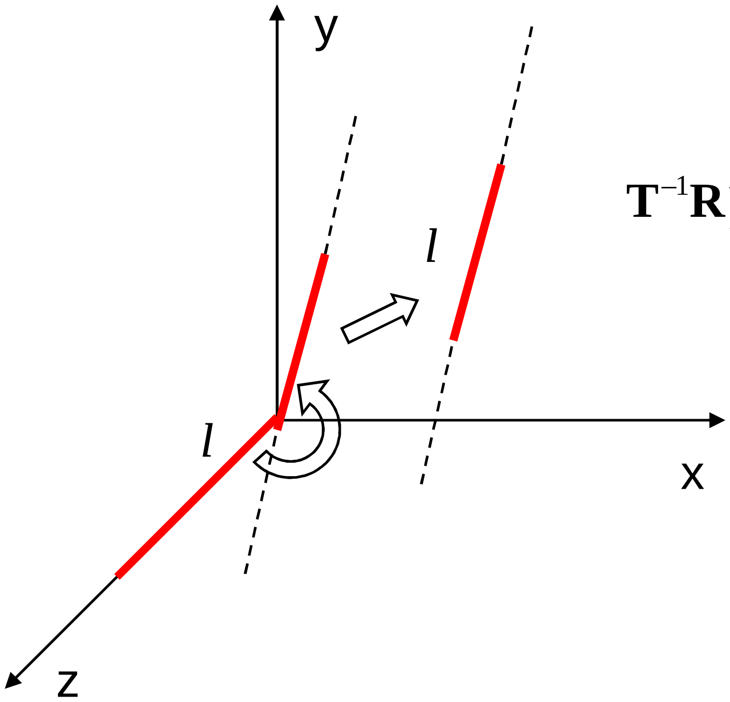
- Step 4. Rotate about *z* axis by the desired angle θ



$$\mathbf{R}_z(\theta)=\begin{bmatrix}\cos\theta & -\sin\theta & 0 & 0\\ \sin\theta & \cos\theta & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

Arbitrary Axis Rotation

- Step 5. Apply the reverse transformation to place the axis back in its initial position



$$\mathbf{T}^{-1}\mathbf{R}_x^{-1}(\alpha)\mathbf{R}_y^{-1}(\beta)=\begin{bmatrix}1&0&0&-x_1\\0&1&0&-y_1\\0&0&1&-z_1\\0&0&0&1\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&\cos\alpha&\sin\alpha&0\\0&-\sin\alpha&\cos\alpha&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}\cos\beta&0&\sin\beta&0\\0&1&0&0\\-\sin\beta&0&\cos\beta&0\\0&0&0&1\end{bmatrix}$$

$$\mathbf{R}(\theta)=\mathbf{T}^{-1}\mathbf{R}_x^{-1}(\alpha)\mathbf{R}_y^{-1}(\beta)\mathbf{R}_z(\theta)\mathbf{R}_y(\beta)\mathbf{R}_x(\alpha)\mathbf{T}$$

3D Rotation with quaternions

Quaternions:

- Extension of complex numbers
- Four components
- Scalar value + 3D vector: $q=(s,\mathbf{v})$
- Special calculation rules
- Compact description of rotations
- To be used if many complex rotations have to be done (esp. animation)

3D scaling

Scale with factors s_x, s_y,s_z :

$$x' = s_x\, x,\;\; y' = s_y\, y,\;\; z' = s_z\, z$$

or

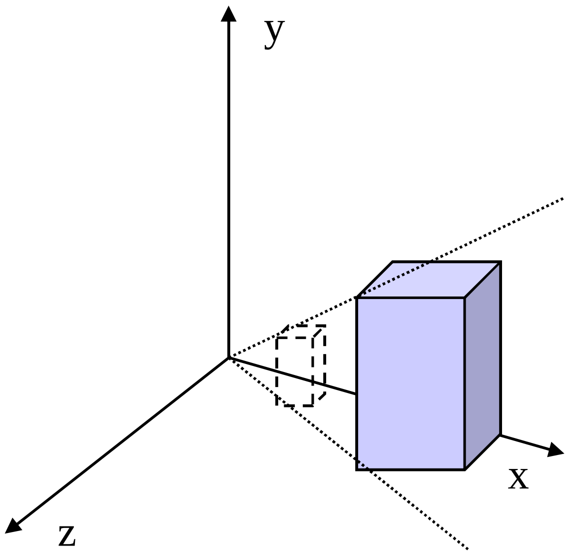
$$\mathbf{P}' = \mathbf{SP}, \text{ or}$$

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

3D Scaling

- Uniform Scaling

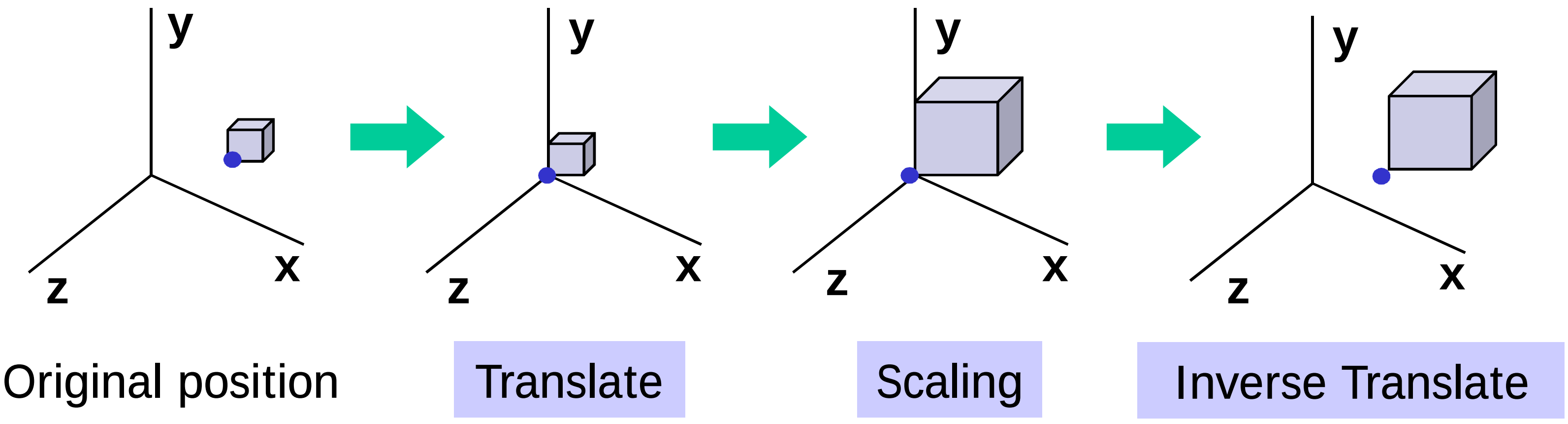
$$x' = x \cdot s_x\,,\quad y' = y \cdot s_y\,,\quad z' = z \cdot s_z$$



$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Relative Scaling

- Scaling with a Selected Fixed Position



$$T(x_f,y_f,z_f)\cdot S(s_x,s_y,s_z)\cdot T(-x_f,-y_f,-z_f)=\begin{bmatrix}x'\\y'\\z'\\1\end{bmatrix}=\begin{bmatrix}1&0&0&x_f\\0&1&0&y_f\\0&0&1&z_f\\0&0&0&1\end{bmatrix}\begin{bmatrix}s_x&0&0&0\\0&s_y&0&0\\0&0&s_z&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}1&0&0&-x_f\\0&1&0&-y_f\\0&0&1&-z_f\\0&0&0&1\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$$

More 3D transformations

+/- same as in 2D:

- Matrix concatenation by multiplication
- Reflection
- Shearing
- Transformations between coordinate systems

Affine transformations 1

Generic name for these transformations:
affine transformations

$$x' = a_{xx}x + a_{xy}y + a_{xz}z + b_x$$

$$y' = a_{yx}x + a_{yy}y + a_{yz}z + b_y$$

$$z' = a_{zx}x + a_{zy}y + a_{zz}z + b_z$$

Affine transformations 2

Properties:

1. Transformed coordinates x',y' and z' are *linearly* dependent on original coordinates x, y en z.
2. Parameters a_{ij} en b_k are constant and determine type of transformation;
3. Examples: translation, rotation, scaling, reflection
4. Parallel lines remain parallel
5. Only translation, rotation reflection: angles and lengths are maintained

Thank You….